Reconstruction of Flows of Mass under the L1 Metric

Jörg Bernhardt* Stefan Funke† Sabine Storandt†

Abstract

We consider the problem of 'moving mass' under the L1 metric. In contrast to the well-known Earth-Mover's-Distance (EMD, [5]) we are not only interested in the amount of work (aka EM distance) that is necessary to transfer the mass but also in reconstructing (local) translations that lead to this movement of mass. This problem is motivated by an image matching application for analysis of electrophorese gel experiments in biology.

1 Introduction

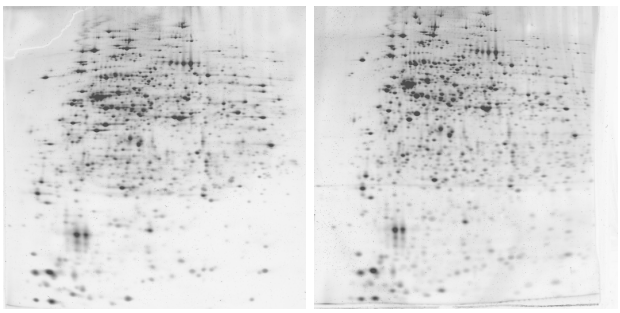

Figure 1: Gel scans for Bacillus Subtilis where we are interested in corresponding spots in both scans.

In the last decade *proteomics* – a branch of molecular biology – has been established as an important area of research. The goal is to detect the totality of proteines – the proteome – inside a cell and identify and quantify single proteins in order to relate them to their functions. For that purpose comparing the proteome of different samples is the method of choice. For example bacteria that suffer from starvation are expected to produce less proteines which are not essential (e.g. for cell division), but instead more proteines that work as enzymes for alternative food sources. In order to detect which protein belongs to which group we have to compare the sample of the starving bacteria to the one of a control group. To that end electrophoresis is a common and cheap method to separate the up to several thousands of proteins that might occur in a cell. Here, in the first dimension isoelectric focusing is used, essentially ordering the proteins according to their pH values. In the second dimension the proteins are then separated according to their masses. The result is a two-dimensional separation of the proteins visualized in a *gel scan*, where proteins can be seen as dark accumulations – so called spots (depicted in Figure 1). The size and the color depth of a spot is proportional to the produced amount of protein. The main difficulty is to identify corresponding *protein spots* in both gel scans. Ideally, if the separation in both dimensions was perfect, each protein would end up at the same position in both scans. Unfortunately, the separation steps are physical processes which are easily disturbed resulting in global *and* local distortions. Hence the identification of those correspondences is highly non-trivial.
To make this problem accessible for mathematical methods, we make the following assumptions:

1. The two separation steps of the gel electrophoresis are independent. Therefore the distance between spots and hence pixels should reflect this characteristic. This makes the L_1 metric most appropriate in our application

2. Very long distances between corresponding spots in the two images are unlikely. So a first approach should be to find a solution there the sum of distances of corresponding spots is minimized.

3. During the protein separation no inversions should occur in the first or the second dimension. In practice this is typically true in the first dimension but sometimes might be violated in the second dimension due to local irregularities in the gel substance which causes proteins to travel at different speeds at different places of the gel. Therefore a second optimization goal should be to find a solution with a low number of inversions, especially on a local scale.

The well-known *Earth Mover's Distance* is able to capture the conditions 1. and 2., while for the third assumption further algorithmic care is needed.

Related Work

The earth mover's distance was proposed in [5], but there and in the subsequent papers it was employed as a pure *distance measure*. The actual transformations that lead to the distance were not really used

*Decodon GmbH, D-17489 Greifswald, bernhardt@decodon.com

†Institut für Formale Methoden der Informatik, Universität Stuttgart, D-70569 Stuttgart,{stefan.funke, sabine.storandt}@fmi.uni-stuttgart.de

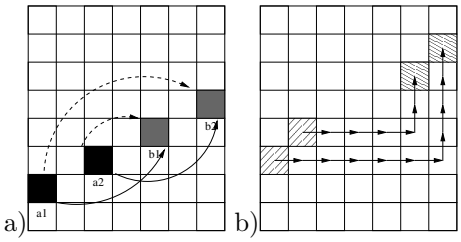

Figure 2: Shortcomings of simple EMD formulations.

or even derived. For the gel matching problem there are several approaches like [3] or [1] which fundametally differ from ours, though, as they first perform a spot detection and then a try to match corresponding spots. Hence they are more susceptible to the employed mathematical spot model.

2 Computation and Reconstruction of EMD Flows

2.1 Naive EMD Computation

EMD We are given two sets of points $A, B \subset \mathbb{R}^2$ and a weight function $w : A, B \to \mathbb{R}$. The term *Earth Mover's Distance* as coined by Rubner et al. [5] already captures its intuitive meaning. We consider the weight of points in set $A \subset \mathbb{R}^2$ as mass that needs to be moved to the locations given by set $B \subset \mathbb{R}^2$. We can compute the EMD via the following linear program:

$$\begin{aligned} min \quad & \sum_{a\in A, b\in B} f_{ab} d(a,b) && (1)\\ \forall a \in A \quad & \textstyle\sum_{b'\in B} f_{ab'} = w(a)\\ \forall b \in B \quad & \textstyle\sum_{a'\in A} f_{a'b} = w(b)\\ \forall a \in A, b \in B \quad & f_{ab} \geq 0\\ \sum_{a\in A, b\in B} f_{ab} \quad & = \min(\textstyle\sum_{a\in A} w(a), \sum_{b\in B} w(b)) \end{aligned}$$

Here, $f_{ab} = c$ denotes that c units of mass are moved from $a \in A$ to the location $b \in B$. From now on we refer to them as 'flows'. Moreover $d(a, b)$ describes the 'cost' of moving one unit of mass from a to b, in our case $d(a, b) = ||a - b||_1 = \sum_{i=1}^{n} |a_i - b_i|$. $w(a)/w(b)$ denotes the amount of supply/demand of mass at location $a \in A$, $b \in B$, respectively.

In general there are several optimal solutions of the EMD-LP. Note, that this is not specific for the L_1 metric but can also occur then using other distances like the euclidean metric, as depicted in figure 2 a). Here, the bold assignment would be the most reasonable one in our scenario, but the dashed correspondences lead to the same objective value under the L_1 as well as the L_2 metric. To get the EMD value, defined as $EMD(A, B) = (\sum_{a\in A} \sum_{b\in B} f_{ab} d(a, b))/(\sum_{a\in A, b\in B} f_{ab})$, this is not of any interest. But our goal now is to identify the solution that fits best to for our envisioned application scenario.

For sake of simplicity we restrict in the following to the case of *binary* images. Our proposed algorithm also generalizes in a straightforward manner to grayscale images (with mass > 1 at a single pixel), the proof for the running time gets more involved, though, introducing a dependency on the total mass.

We define the *Earth Mover's Corresponding Problem (EMCP)* as follows:

Definition 1 EMCP: *Given two sets $A, B \subset \mathbb{R}^2$. Compute the/a function $\phi : A \to B$ that minimizes the number of local inversions while maintaining an optimal solution for the corresponding EMD-LP.*

Essentially the EMCP is the problem of computing a specific *matching* in a geometric bipartite graph.

2.2 A Compact EMD Formulation for L_1

The size of the naive EMD formulation is essentially $\Theta(|A| \cdot |B|)$ which makes this formulation rather useless for real world instances from our application domain, e.g. for gel scans of around 1 Megapixel each with around 15% 'mass' we would already deal with more than 22 billion variables. In case of the L_1 metric, there is a simple and considerably more compact formulation[1]. Let us assume that our points in sets A and B have integer coordinates (like the pixel coordinates of two images). The idea is to decompose a flow from (x, y) to (x', y') with $x \leq x', y \leq y'$ into a sequence of $|x - x'|$ horizontal 'miniflows' $(x, y) \to (x + 1, y) \to (x + 2, y) \to \cdots \to (x', y)$ followed by a sequence of $|y - y'|$ vertical miniflows $(x', y) \to (x', y + 1) \ldots$. The respective LP formulation now has the following structure:

$$min \sum f_{xyx'y'}$$

$$\begin{aligned} a_{xy} + f^h_{(x-1)yxy} - f^h_{xy(x+1)y} - f^h_{xy(x-1)y} + f^h_{(x+1)yxy} &= E_{xy}\\ E_{xy} + f^v_{x(y-1)xy} - f^v_{xyx(y+1)} - f^v_{xyx(y-1)} + f^v_{x(y+1)xy} &= b_{xy}\\ E_{xy} &\geq 0\\ f_{xyx'y'} &\geq 0 \end{aligned}$$

Here, a_{xy} denotes the mass of point set A at position (x, y), the same for b_{xy}, respectively. $f^{h/v}_{xyx'y'}$ denotes the horizontal/vertical flow from pixel (x, y) to a (neighboring) pixel (x', y') (which now can take any positive value). E_{xy} can be interpreted as the intermediate distribution of mass after the horizontal shifts. Any feasible solution to the original formulation can be translated into this more compact formulation and vice versa as we will see in the following. The size of this formulation is *linear* in the size of the gel scan and its solution can be found using some standard LP solver like CPLEX [2] or specialized network

[1] This formulation has probably been used before but we could not come up with a reference.

simplex implementations like mcf [4]. Note that it is easy to handle sets/images with different masses by always requiring A to be the smaller set and replacing the second equality by an inequality.

2.3 Preliminary Flow Reconstruction

Having computed an optimal solution to the compact LP, our first goal is to construct actual flows of mass from set A to set B. We will proceed in two steps: first the horizontal miniflows are turned into horizontal flows, then, in a second step vertical flows are constructed from the intermediate mass distribution and the miniflows. Finally, in a last step, the horizontal and vertical flows are combined to flows for the original problem formulation.

1-dimensional Reconstruction: Essentially we first solve a 1-dimensional EMCP in each row of the image. All horizontal flows, the intermediate mass distributions and hence the excesses and deficits are known. We then sweep from left to right, balancing excesses and deficits in a first-in-first-out fashion. It is easy to see that this procedure balances the deficits and excesses at minimal cost, solving a 1-dimensional EMCP. Exactly the same procedure is applied vertically in each column to distribute the mass particles from the intermediate mass distribution to their final locations according to set B. It remains to combine the horizontal and vertical flows to obtain an actual matching between pixels in the source and the target image.

Gluing together Horizontal and Vertical Flows: At this point we can arbitrarily combine horizontal and vertical flows to obtain a cost optimal matching. In our implementation we employ a simple greedy strategy which combines incoming horizontal and outgoing vertical flows to diagonal flows.

While the resulting matching always yields flows which are optimal wrt the Earth Mover's distance, often this reconstruction does not lead to the 'correct' flows in our application. The problem is that the miniflows we get from a solution to LP (2.2) might not even allow for a reconstruction of the correct flows, see Figure 2 b).

2.4 Local Search on Preliminary Flows

Now we want to solve the *EMCP* based on our EMD solution. For real instances of our application we can not expect that a completely inversion-free solution exists. Nevertheless as a sanity check and for theoretical soundness it is desirable that our algorithm recovers a inversion free matching if such exists. We refer to sets allowing an inversion free matching from now on as *well-sorted*.

Definition 2 (well-sortedness) *We call two sets of points $A, B \subset \mathbb{R}^2$ with $|A| = |B|$* well-sorted *if there exists a bijective function/transformation/matching $\phi : A \to B$ with $\cup_{a\in A}\phi(a) = B$ and $\forall a, a' \in A$*

$$a_1 \lessgtr a'_1 \Rightarrow \phi(a)_1 \lessgtr \phi(a')_1 \quad \textit{and}$$
$$a_2 \lessgtr a'_2 \Rightarrow \phi(a)_2 \lessgtr \phi(a')_2$$

We note that for well-sorted sets A and B, the transformation ϕ that we want to recover and which witnesses the well-sortedness has optimal cost, i.e. has cost equal to the Earth Mover's distance (this actually requires proof).

After glueing together the miniflows we obtain a set of flows which are typically not well-sorted. Our idea now is to consider flows pairwise and exchange targets in case this improves 'well-sortedness'. Interestingly we can show that if the two sets A and B are well-sorted this leads to a well-sorted matching after $O(n^2)$ iterations. But even if A and B were not well-sorted, this procedure terminates and yields a drastically improved assignment. We denote by $y(f)$ for a flow $f = (a, b)$ its extent in y direction.

Definition 3 (Switch) *Let $f_1 = (a, b)$ and $f_2 = (a', b')$ be a pair of flows and $f_1^s = (a, b'), f_2^s = (a', b)$ their equivalents after switching the targets. A switch is feasible if*

$$\begin{aligned}
&1.)\ cost(f_1) + cost(f_2) = cost(f_1^s) + cost(f_2^s) \\
&2.a)\ \max(y(f_1), y(f_2)) > \max(y(f_1^s), y(f_2^s)) \qquad \textit{or} \\
&2.b)\ \max(y(f_1), y(f_2)) = \max(y(f_1^s), y(f_2^s)) \qquad \textit{and} \\
&\quad\ \max(x(f_1), x(f_2)) > \max(x(f_1^s), x(f_2^s))
\end{aligned}$$

We call case 2.a) a *height decreasing*, case 2.b) a *width decreasing* switch.

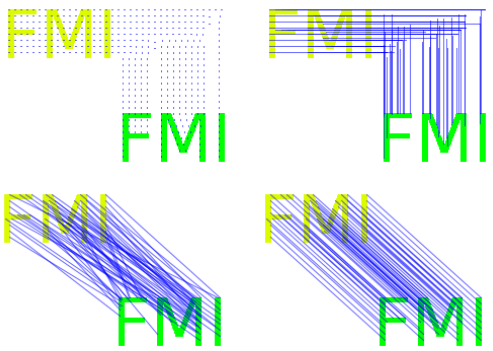

Figure 3: Reconstruction of a global translation: LP-solution in miniflows, 1-dimensional flows, greedy diagonal flows, after local search.

2.4.1 Algorithm

It is convenient to ensure that all y-coordinates are different and in particular, all heights of flows are pairwise different. We ensure this by standard techniques for symbolic perturbation on the y-coordinates. Our basic two-phase local search algorithm is then simple to state:

1. perform height decreasing switches between flows f_a and f_b as long as possible
2. perform width decreasing switches between flows f_a and f_b as long as possible

2.4.2 Runtime and Correctness

The running time of our switching algorithm crucially depends on the order in which we perform switches. For the height decreasing switches, we order the flows according to the y-coordinate of their sources and always check for the 'smallest' flow according to this order and switch it with the next smallest switchable flow. In the same manner we perform the width decreasing switches (ordered by x-coordinate).

Lemma 1 *For any a, a', flows associated with a and a' get switched at most once in phase 1 and at most once in phase 2 if we always use above criterion for selecting the next switch pair.*

This Lemma immediately yields a bound of $O(n^2)$ on the number of switches. A naive implementation can process one switch in time $O(n)$. For correctness, two core Lemmas have to be proven:

Lemma 2 *Let ϕ^* be a transformation witnessing the well-sortedness of sets A and B. Then ϕ^* does not allow any height or width decreasing switches.*

Lemma 3 *Let ϕ be the current set of flows, ϕ not witnessing well-sortedness of well-sorted sets A and B, then there exists a height or width decreasing switch.*

Particularly proving the latter Lemma is involved and requires a a closer examination of the structure of flows not witnessing well-sortedness. Using these Lemmas we obtain the our main theorem:

Theorem 4 *For well-sorted sets A and B our algorithm which starts with a cost-optimal solution terminates after $O(n^3)$ time and recovers a matching ϕ which witnesses well-sortedness of A and B.*

For not perfectly well-sorted instances correctness seems hard to formalize. From a biological point of view the solution of the EMCP is meaningful if we can assure well-sortedness at least for flows that are close to each other. It is not clear whether a further exploration of formal models for the 'correct' solution is worthwhile; as usual, the precise formalization of real-world problems seems challenging.

3 Experiments

In Figure 3 we have depicted the main steps of our approach. We compared the results of our algorithm to other approaches (like weighted bipartite matching) on model data. We always achieved the solution with the minimal number of inversions as well as the highest rate of correct correspondencies. Furthermore we checked on real data if our solution comes close to the so called manual edits (see figure 4), that could be seen as ground truth. Although we worked only on the binary versions of the image scans, we reconstructed about 84% (average) of the flows correctly.

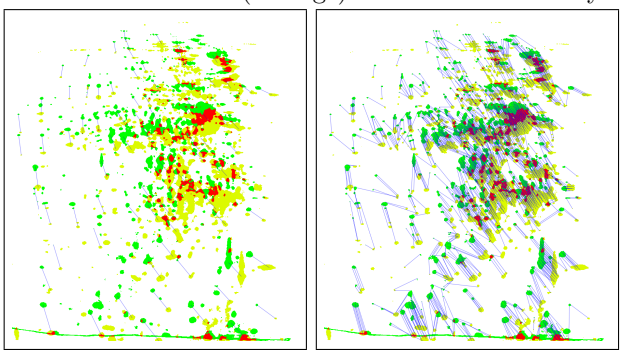

Figure 4: Comparison of a manual edit (left) and the flows resulting from our algorithm (middle, subsampled). Green: Image A, Yellow: Image B, Red:Overlap.

4 Conclusion

We presented an algorithm which faithfully determines correspondencies between protein spots in our concrete application from computational biology. Due to the nature of the physical process, those problem instances are typically not well-sorted but somewhat 'close' to being well-sorted. While truly well-sorted instances could be trivially solved by a simple sorting procedure, our algorithm also works for those real-world instances – the sorting approach does not. For well-sorted instances, we can prove our algorithm to be correct. Other application domains where this kind of matching could be of interest are in the field of vision or video compression.

References

[1] M. Berth, F. Moser, M. Kolbe, and J. Bernhardt. The state of the art in the analysis of 2-dimensional gel electrophoresis images. *Appl Microbiol Biotechnol*, 2007.

[2] R. E. Bixby. Implementing the simplex method: The initial basis. *INFORMS Journal on Computing*, 4(3):267–284, 1992.

[3] A. Efrat, F. Hoffmann, K. Kriegel, C. Schultz, and C. Wenk. Geometric algorithms for the analysis of 2d-electrophoresis gels. *Journal of Computational Biology*, 9(2):299–315, 2002.

[4] A. Loebel. Mcf version 1.3 - a network simplex implementation. available for academic use free of charge. `http://www.zib.de`, 2004.

[5] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *Int. J. Comput. Vision*, 40(2):99–121, 2000.